20

A Regime Recognition Algorithm for Helicopter Usage Monitoring

David He[1], Shenliang Wu[1] and Eric Bechhoefer[2]
[1]*The University of Illinois at Chicago*
[2] *Goodrich*
USA

1. Introduction

The importance of regime recognition for structural usage monitoring of helicopters cannot be overemphasized. Usage monitoring entails determining the actual usage of a component on the aircraft. This allows the actual usage/damage from a flight to be assigned to that component instead of the more conservative worst-case usage. By measuring the actual usage on the aircraft, the life of the components can be extended to their true lifetime. Usage monitoring requires an accurate recognition of regimes, where a regime is the flight profile of the aircraft at each instant of the flight. For each regime, a damage factor is assigned to each component that has usage. These damage factors are assigned by the original equipment manufactures (OEMs) based on measured stresses in the aircraft when undergoing a given maneuver. Therefore, it is important that the regimes can be recognized correctly during the flight of the aircraft to avoid either underestimated or overestimated damages for the aircraft. Another important aspect of regime recognition is related to the certification of health usage and monitoring system (HUMS). As outlined in a document of the Federal Aviation Administration (FAA) HUMS R&D Initiatives (Le, 2006), regime recognition and monitoring has been identified as a high priority HUMS R&D short-term task in the area of structural usage monitoring and credit validation. The certification readiness and the aircraft applicability of regime recognition and credit validation are lower in comparison to the overall HUMS assessment (12% to 18% and 59% to 82%, respectively). These cited assessment results clearly show the weakness of current regime recognition methods in HUMS.

Although important, not much work on regime recognition has been published. Two recent research papers are worth mentioning here. The first paper (Teal *et al.*, 1997) described a methodology for mapping aircraft maneuver state into the MH-47E basic fatigue profile flight regimes in a manner which ensures a conservative, yet realistic, assessment of critical component life expenditure. They also presented the use of wind direction and magnitude estimation and inertial/air data blending to obtain high fidelity airspeed estimation at low speeds. An accuracy rate of 90% based on time was reported. This method basically is a logical test. The system firstly identifies the maneuver based on flight dynamic data and general principles of tandem rotor helicopter flight which are derived from flight experience and mathematical models correlated with flight test results, then the aircraft maneuver state is mapped directly into one of the basic fatigue profile flight regimes. The method is subject

Source: Aerospace Technologies Advancements, Book edited by: Dr. Thawar T. Arif,
ISBN 978-953-7619-96-1, pp. 492, January 2010, INTECH, Croatia, downloaded from SCIYO.COM

to the main weakness of logical test in dealing with the noisy measurement. If the measured parameters were free of noise, logical tests would give accurate results. The second paper (Berry *et al*., 2006) presented a regime recognition scheme implemented as a hierarchical set of elliptical function (EBF) neural networks. Motivated to develop an automatic regime recognition capability as an enhancement to the US Army's Vibration Management Enhancement Program (VMEP), the EBF neural networks were devised to simplify neural net training and to improve the overall performance. The idea of using a hierarchical set of neural networks is to group individual regimes into regime groups, including an unknown regime group (regimes that cannot be classified as any regimes in one of the known regime groups). Regimes in each group are classified by an individual net in the hierarchical set. Regime recognition is carried out through a hierarchical process, e.g., if a regime cannot be classified as the first regime group by the top net in the hierarchy it will be passed to the lower level nets for further classification. In the paper, a total of 141 regimes of Sikorsky's S-92 helicopter were grouped into 11 groups, including "level flight", "auto", "climb", "dive", and etc. As shown in the paper, the EBF neural network regime recognition scheme gave near perfect classification results for "level flight" regimes. However, the results for classification of all regime groups didn't show a consistent effectiveness of the scheme. For example, for "level flight" group the classification rate is 97.85% but it is 33.18% for "turns" group. Because of the low classification rates for some groups, the scheme gave an overall rate of 76.21%. In addition to the requirement for a large amount of data to train a neural network, one variable that could also affect the performance of the scheme is the way by which the regimes are grouped. Another limitation of neural network is that as it is a black-box methodology, little analytical insights can be gained to enhance the regime recognition process.

Regime recognition is basically a data mining problem, i.e., mining the measured parameter data and mapping them to a defined flight profile. In this paper, the philosophy of data mining is adopted for regime recognition. In particular, a regime recognition algorithm developed based on hidden Markov model (HMM) is presented.

2. Regime recognition algorithm

Before presenting the data mining based approach for regime recognition, we first describe the regime recognition problem from a data mining perspective as following. Suppose we have Q regimes, denoted as $\omega = \{\omega_1, \omega_2, ..., \omega_i, ..., \omega_Q\}$. By taking into account the time factor in regime recognition, each individual regime at time t is expressed as $s(t) = \omega_i$. Given an observation sequence $R = R_1 R_2 ... R_t ... R_T$, where T is the length of observations in the sequence and each observation R_t is a $1 \times O$ vector, denoted as $R_t = \{f_{t1}, f_{t2}, ..., f_{tj}, ... f_{tO}\}^T$ with f_{tj} being the value of feature j of the tth observation and O the number of the features, the objective is to identify regime sequence denoted as $\Omega = \{s(1), s(2), ..., s(T)\}$.

Accordingly, a hidden Markov model $\lambda = (\pi, A, B)$ for regime recognition could be characterized as follows:

1. The initial regime distribution $\pi = \{\pi_i\}$, where $\pi_i = P[s(0) = \omega_i]$, $1 \le i \le Q$.
2. The regime transition probability distribution $A = \{a_{ij}\}$, where $a_{ij} = P[s(t+1) = \omega_j \mid s(t) = \omega_i]$, $1 \le i, j \le Q$.
3. The observation probability distribution in regime ω_j, $B = \{b_j(t)\}$, where $b_j(t) = P[R_t | s(t) = \omega_j]$, $1 \le j \le Q$, $1 \le t \le T$.

The estimation of $\lambda = (\pi, A, B)$ is a crucial step if we want to compute the probability of a system in regime ω_i based on the estimated HMM model $\hat{\lambda}$. Generally, there are two main stages in regime recognition using HMM. The first stage is the training stage. The purpose of training stage is to estimate the three parameters of the HMM. The estimation of $\lambda = (\pi, A, B)$ is carried out through an iterative learning process of adjusting the model parameters to maximize the probability $P(R_{train} \mid \lambda)$ of producing an observation sequence $R_{train} = R_1 R_2 ... R_{T_1}$, given model λ. Therefore, at the end of training process, we could obtain an estimated HMM model $\hat{\lambda}_i = (\hat{\pi}_i, \hat{A}_i, \hat{B}_i)$ for each regime ω_i. The second stage is the testing stage. The purpose of testing is to calculate the probability of generating the unknown observation sequence, given the estimated model $\hat{\lambda}_i$, $1 \le i \le Q$. Given a testing observation sequence $R_{test} = R_1 R_2 ... R_{T_2}$ and a set of estimated models $\lambda = \left\{ \hat{\lambda}_i = (\hat{\pi}_i, \hat{A}_i, \hat{B}_i),\ 1 \le i \le Q \right\}$, log-likelihood LL of R_t from the observation sequence $R_{test} = R_1 R_2 ... R_{T_2}$ can be computed. Note that, in general, $T_1 \ge T_2$.

2.1 The training stage

The training stage is a process to determine model parameters from a set of training data. *A priori* values of π, *A*, and *B* are assumed and observations are presented iteratively to the model for estimation of parameters. Likelihood maximization is the basic concept behind this estimation procedure. In each iteration, the goal is to maximize the expected log-likelihood, i.e., logarithm of the probability that the model generates the observation sequence. This iterative process continues until the change in log-likelihood is less than some threshold and convergence is declared.

In an HMM, the observation probability is assumed to follow a Gaussian distribution. Although, all of the classical parametric densities are unimodal, many practical problems involve multimodal densities. In our algorithm, a Gaussian mixture model (GMM) is used. Let $X = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_k, ... \mathbf{x}_T]$ be the sample dataset and $\mathbf{x}_k$ is a $O \times 1$ vector. Let *M* be the number of mixture components. There is no definition for the number of mixture components per output distribution and there is no requirement for the number of mixture components to be the same in each distribution. If $M = 1$, it is the unimodal density case. When $M > 1$, a mixture model can be expressed as:

$$p(X) = \sum_{i=1}^{M} w_i h(X \mid \mu_i, \sigma_i) \tag{1}$$

where $p(X)$ is the modeled probability distribution function and w_i is the mixture weight of component *i*. Clearly, $w_1 + w_2 + ... + w_M = 1$, and $0 < w_i < 1$ for all $i = 1, 2, \cdots, M$. $h(X \mid \mu_i, \sigma_i)$ is a probability distribution parameterized by μ_i, σ_i, and can be computed as:

$$h(X \mid \mu_i, \sigma_i) = \frac{e^{-\frac{1}{2}d_i^2}}{(2\pi)^{\frac{O}{2}}\sqrt{|\sigma_i|}} \tag{2}$$

where d_i^2 can be computed as:

$$d_i^2 = \left[d_{i1}^2, d_{i2}^2, ..., d_{ik}^2, ..., d_{iT}^2\right]^t$$

$$d_{ik}^2 = (\mathbf{x}_k - \mu_i)^t \sigma_i^{-1} (\mathbf{x}_k - \mu_i),\ k = 1,2,...,T \tag{3}$$

Once w, μ and σ are determined, $p(X)$ is defined, i.e., the observation probability distribution *B*. So the estimation problem of an HMM model $\lambda = (\pi, A, B)$ is converted to estimate $\lambda = (\pi, A, \boldsymbol{\mu}, \boldsymbol{\sigma}, \mathbf{w})$, where $\boldsymbol{\mu} = \{\mu_1, \mu_2, ..., \mu_M\}$, $\boldsymbol{\sigma} = \{\sigma_1, \sigma_2, ..., \sigma_M\}$, $\mathbf{w} = \{w_1, w_2, ..., w_M\}$. Here, the GMM parameters of each HMM model can be split into two groups: the untied parameters that are Gaussian-specific and the tied parameters that are shared among all the Gaussians of all the HMM states.

Although there is no optimal way of estimating the model parameters so far, local optimal is feasible using an iterative procedure such as the Baum-Welch method (or equivalently the expectation-modification method) (Rabiner, 1989; Levinson *et al.*, 1983), or using gradient techniques (Dempster, 1977). In order to facilitate the computation of learning, three forward-backward variables are defined in the forward-backward algorithm:

1. The probability of the partial observation sequence, $R_1R_2...R_t$, and regime ω_i at time *t*, given model λ: $\alpha_t(i) = P[R_1R_2...R_t, s(t) = \omega_i \mid \lambda]$
2. The probability of the partial observation sequence from *t*+1 to the end, given regime ω_i at time *t* and model λ: $\beta_t(i) = P[R_{t+1}R_{t+2}...R_T \mid s(t) = \omega_i, \lambda]$
3. The probability of being in regime ω_i at time *t*, and regime ω_j at time *t*+1, given model λ and observation sequence R_t, i.e.,

$$\xi_t(i,j) = P[s(t) = \omega_i, s(t+1) = \omega_j \mid R_t, \lambda]$$

We initialize forward variable as $\alpha_{t=0}(i) = 0$, for all $1 \le i \le Q$ at time $t = 0$, then in the forward iteration, we calculate the forward variable $\alpha_t(j)$ by the following equations from t = 1 to t = T:

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{Q} \alpha_t(i) a_{ij}\right] b_j(R_{t+1}),\ \text{where: } 1 \le t \le T-1,\ 1 \le j \le Q \tag{4}$$

In the backward iteration, we compute the backward variable $\beta_t(i)$ and $\xi_t(i,j)$ after initialization $\beta_T(i) = 1$ at time t = T:

$$\beta_t(i) = \sum_{j=1}^{Q} a_{ij} b_j(R_{t+1}) \beta_{t+1}(j),\ \text{where: } t = T-1, T-2, ..., 1,\ 1 \le i \le Q \tag{5}$$

$$\xi_t(i,j) = \frac{\alpha_t(i) a_{ij} b_j(R_{t+1}) \beta_{t+1}(j)}{P(R_{train} \mid \lambda)} = \frac{\alpha_t(i) a_{ij} b_j(R_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^{Q}\sum_{j=1}^{Q} \alpha_t(i) a_{ij} b_j(R_{t+1}) \beta_{t+1}(j)},\ \text{where: } 1 \le i, j \le Q \tag{6}$$

We define $\gamma_t(i)$ as the probability of being in regime ω_i at time *t*, given the observation sequence and the model. Therefore, $\gamma_t(i) = \sum_{j=1}^{Q} \xi_t(i,j)$. So, $\hat{a}_{ij}$, the estimated probability of a regime transition from $s(t-1) = \omega_i$ to $s(t) = \omega_j$, can be calculated by taking the ratio between the expected number of transitions from ω_i to ω_j and the total expected number of any transitions from ω_i can be computed as:

$$\hat{a}_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)}, \; 1 \le i \le Q, \quad 1 \le j \le Q \tag{7}$$

Then the estimated observation probability distribution $(\boldsymbol{\mu}_j, \boldsymbol{\sigma}_j, \mathbf{w}_j)$ in regime ω_j can be computed as:

$$\hat{\boldsymbol{\mu}}_j = \frac{\sum_t \mathbf{w}_t^j E_{tj}[R_{train}]}{\sum_t \mathbf{w}_t^j} \tag{8}$$

$$\hat{\boldsymbol{\sigma}}_j = \frac{\sum_t \mathbf{w}_t^j E_{tj}[R_{train} R_{train}^{'}]}{\sum_t \mathbf{w}_t^j} - \hat{\boldsymbol{\mu}}_j \hat{\boldsymbol{\mu}}_j^{'} \tag{9}$$

Note in equations (8)- (9), weights $\mathbf{w}_t^j = \sum_{t=1}^{T-1} \gamma_t(j)$ are posterior probabilities, $E_{tj}[R_{train}]$ is the mathematical expectation of observation R_{train} at time *t* under regime ω_j, $\hat{\boldsymbol{\mu}}_j, \hat{\boldsymbol{\sigma}}_j$ are both untied parameters of GMM, and the covariance type is full.
Similarly, the estimated initial regime distribution can be computed as:

$$\hat{\pi}_i = \gamma_1(i), \; 1 \le i \le Q \tag{10}$$

2.2 The testing stage

As mentioned before, testing is a stage to evaluate the likelihood of an unknown observation belonging to a given regime. Since model $\hat{\lambda}_i = (\pi_i, A_i, \boldsymbol{\mu}_i, \boldsymbol{\sigma}_i, \mathbf{w}_i)$ has been built up in the training process, it can be used to calculate the log-likelihood *LL* of a testing observation R_{test} based on model $\hat{\lambda}_i$ for regime ω_i. This log-likelihood can be calculated efficiently using the forward algorithm.
The probability that model $\hat{\lambda}_i$ produces observation R_{test} is computed as:

$$P(R_{test} \mid \hat{\lambda}_i) = \alpha(i)$$

By definition, $\alpha(i)$ is the probability of generating R_{test} and ending in regime ω_i, therefore,

$$P(R_{test} \mid \hat{\lambda}_i) = \alpha(i) = P[s(t) = i] P[R_{test} \mid s(t) = i] = \pi_i p_i(R_{test}) \quad 1 \le i \le Q \tag{11}$$

In (11), $p_i(R_{test})$ can be calculated from equation (1), and R_{test} in regime recognition is the unknown signal. So, the log-likelihood value is computed as:

$$LL_i = \mathrm{Ln}[P(R_{test} \mid \hat{\lambda}_i)] \tag{12}$$

To classify a testing observation into one of Q regimes, train Q HMMs, one per regime, and then compute the log-likelihood that each model gives to the testing observation, a set of log-likelihood value $LL = \{LL_1, LL_2, ..., LL_i, ..., LL_Q\}$ will be obtained.

3. Algorithm validation

In this section, the developed regime recognition algorithm was validated using the Army UH-60L flight card data.

The Army UH-60L flight card data was collected during a flight test and provided by Goodrich. The intent of the flight test was to provide flight data which could be used to refine and revise a preliminary set of regime recognition algorithms. The test pilots annotated detailed flight cards with actual event times as maneuvers were conducted during the UH-60L regime recognition flights. The on-board pilots maintained a detailed log of the maneuvers, flight conditions, and corresponding event times encountered during the mission flight. A total of 50 regimes were conducted with annotation in the flight test. A limited amount of non-annotated actual flight data was used prior to the flight test to check the functionality of the HUMS system. The recorded data was downloaded and processed after the flight test.

For the Army UH-60L helicopter, a total of 90 preliminary regimes were defined by original equipment manufacturer (OEM). Data of 22 basic aircraft parameters were collected from sensors mounted on the aircraft, or sensors added to the Goodrich IMD-HUMS system for regime recognition. These parameter data is used for the identification of events, control reversals, and regimes. The parameter monitoring is performed during the whole ground-air-ground (GAG) cycles, from rotor start to rotor shutdown, and takeoff to landing. Table 1 provides the list of parameters with their description collected from IMD-HUMS system for regime recognition.

During the validation process, the dataset was randomly divided into two subsets: 70% of data was used for training and 30% for testing. By using the training data, an HMM model was built for each regime. Then the testing data was input into the trained HMM models to compute the log-likelihood values. The maximum log-likelihood value indicates the identified regime. The confusion matrix generated during the testing is provided in Table 2. From the results in Table 2, we see that the overall accuracy of the regime recognition is 99%.

Note that in solving regime recognition using HMM, the training set is dependent on the time sequence of maneuvers. Thus, it is able to find the regime or very complex grouped maneuvers. On the down side, the training set is really too small to capture all of the various maneuvers sequences that could be encountered. For example, it likely that from straight and level, you could go in to a left turn, or a right turn. From a right turn, you can go back to level, climbing right turn, diving right turn, or a higher angle of back turn. In reality, a flight card should contain all of the mixed mode maneuvers.

Parameter No.	Parameter Name	Parameter Description
1	WowDly	WOW Delayed
2	LngFlg	Landing Flag
3	TkOFlg	Takeoff Flag
4	RollAt	Roll Attitude
5	PtchAt	Pitch Attitude
6	RdAlt	Radar Altitude
7	YawDt	Yaw Rate
8	AltDt	Altitude Rate
9	LatDt2	Lateral Acceleration
10	VertAccl	Vertical Acceleration
11	MrRpm	RPM of Main Rotor
12	CrNz	Corrected Normal Acceleration
13	CalSpd	Calibrated Airspeed
14	Vh	Airspeed Vh Fraction
15	TGT	Turbine Gas Temperature
16	RMS_Nz	RMS Normal Acceleration
17	TEngTrq	Torque 1/Torque 2
18	AOB	Angle of Bank
19	CR_Pedal	Control Reversal Flag
20	Cr_Colct	Corrected Collective Rate
21	Cr_Lat	Corrected Latitude
22	Cr_Lon	Corrected Longitude

Table 1. Monitored parameters in IMD-HUMS system

Regime	2	3	4	5	7	8	9	10	11	12	13	14	15	16
2	1	1	1	1	1	1	1	1	1	1	1	1	1	1
3	1	0.99	1	0.99	1	1	1	1	1	1	1	1	1	1
4	1	1	0.99	1	1	1	1	1	1	1	1	1	1	1
5	0.05	0.66	0.26	0.97	1	1	1	1	1	1	1	1	1	1
7	1	1	1	1	0.99	0.5	1	1	1	1	1	1	1	1
8	1	1	1	1	0.95	1	1	1	1	1	1	1	1	1
9	1	1	1	1	1	1	0.99	1	1	1	1	1	1	1
10	1	1	1	1	1	1	0.97	1	0.98	1	0.93	0.99	1	1
11	1	1	1	1	1	1	1	1	1	0.93	0.91	0.98	1	1
12	1	1	1	1	1	1	1	1	1	0.99	0.87	0.96	1	1
13	1	1	1	1	1	1	1	1	1	0.99	1	1	1	1
14	1	1	1	1	1	1	1	1	1	1	1	0.96	1	1
15	1	1	1	1	1	1	1	1	1	1	1	1	0.98	1
16	1	1	1	1	1	1	1	1	1	1	1	1	0.93	0.9
17	1	1	1	1	1	1	1	1	1	1	1	1	0.98	1
19	1	1	1	1	1	1	1	1	1	1	1	1	1	1
20	1	1	1	1	1	1	1	1	1	1	1	1	1	1
21	1	1	1	1	1	1	1	1	1	1	1	1	1	1
22	1	1	1	1	1	1	1	1	1	1	1	1	1	1
23	1	1	1	1	1	1	1	1	1	1	1	1	1	1
24	1	1	1	1	1	1	1	1	1	1	1	1	1	1
25	1	1	1	1	1	1	1	1	1	1	1	1	1	1
26	1	1	1	1	1	1	1	1	1	1	1	1	1	1
27	1	1	1	1	1	1	1	1	1	1	1	1	1	1
28	1	1	1	1	1	1	1	1	1	1	1	1	1	1
36	1	1	1	1	1	1	1	1	1	1	1	1	1	1
37	1	1	1	1	1	1	1	1	1	1	1	1	1	1
40	1	1	1	1	1	1	1	1	1	1	1	1	1	1
41	1	1	1	1	1	1	1	1	1	1	1	1	1	1
42	1	1	1	1	1	1	1	1	1	1	1	1	1	1
43	1	1	1	1	1	1	1	1	1	1	1	1	1	1
44	1	1	1	1	1	1	1	1	1	1	1	1	1	1
45	1	1	1	1	1	1	1	1	1	1	1	1	1	1
46	1	1	1	1	1	1	1	1	1	1	1	1	1	1
48	1	1	1	1	1	1	1	1	1	1	1	1	1	1
49	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Table 2. The confusion matrix of validation test

Regime	2	3	4	5	7	8	9	10	11	12	13	14	15	16
50	1	1	1	1	1	1	1	1	1	1	1	1	1	1
51	1	1	1	1	1	1	1	1	1	1	1	1	1	1
52	1	1	1	1	1	1	1	1	1	1	1	1	1	1
53	1	1	1	1	1	1	1	1	1	1	1	1	1	1
54	1	1	1	1	1	1	1	1	1	1	1	1	1	1
55	1	1	1	1	1	1	1	1	1	1	1	1	1	1
56	1	1	1	1	1	1	1	1	1	1	1	1	1	1
57	1	1	1	1	1	1	1	1	1	1	1	1	1	1
59	1	1	1	1	1	1	1	1	1	1	1	1	1	1
60	1	1	1	1	1	1	1	1	1	1	1	1	1	1
61	1	1	1	1	1	1	1	1	1	1	1	1	1	1
63	1	1	1	1	1	1	1	1	1	1	1	1	1	1
64	1	1	1	1	1	1	1	1	1	1	1	1	1	1
65	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Table 2. The confusion matrix of validation test (Continued 1)

Regime	41	42	43	44	45	46	48	49	50	51	52	53	54	55
2	1	1	1	1	1	1	1	1	1	1	1	1	1	1
3	1	1	1	1	1	1	1	1	1	1	1	1	1	1
4	1	1	1	1	1	1	1	1	1	1	1	1	1	1
5	1	1	1	1	1	1	1	1	1	1	1	1	1	1
7	1	1	1	1	1	1	1	1	1	1	1	1	1	1
8	1	1	1	1	1	1	1	1	1	1	1	1	1	1
9	1	1	1	1	1	1	1	1	1	1	1	1	1	1
10	1	1	1	1	1	1	1	1	1	1	1	1	1	1
11	1	1	1	1	1	1	1	1	1	1	1	1	1	1
12	1	1	1	1	1	1	1	1	1	1	1	1	1	1
13	1	1	1	1	1	1	1	1	1	1	1	1	1	1
14	1	1	1	1	1	1	1	1	1	1	1	1	1	1
15	1	1	1	1	1	1	1	1	1	1	1	1	1	1
16	1	1	1	1	1	1	1	1	1	1	1	1	1	1
17	1	1	1	1	1	1	1	1	1	1	1	1	1	1
19	1	1	1	1	1	1	1	1	1	1	1	1	1	1
20	1	1	1	1	1	1	1	1	1	1	1	1	1	1
21	1	1	1	1	1	1	1	1	1	1	1	1	1	1
22	1	1	1	1	1	1	1	1	1	1	1	1	1	1
23	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Table 2. The confusion matrix of validation test (Continued 2)

Regime	41	42	43	44	45	46	48	49	50	51	52	53	54	55
24	1	1	1	1	1	1	1	1	1	1	1	1	1	1
25	1	1	1	1	1	1	1	1	1	1	1	1	1	1
26	1	1	1	1	1	1	1	1	1	1	1	1	1	1
27	1	1	1	1	1	1	1	1	1	1	1	1	1	1
28	1	1	1	1	1	1	1	1	1	1	1	1	1	1
36	1	1	1	1	1	1	1	1	1	1	1	1	1	1
37	1	1	1	1	1	1	1	1	1	1	1	1	1	1
40	1	1	1	1	1	1	1	1	1	1	1	1	1	1
41	1	1	1	1	1	1	1	1	1	1	1	1	1	1
42	1	0.99	1	1	1	1	1	1	1	1	1	1	1	1
43	1	1	0.99	1	1	1	1	1	1	1	1	1	1	1
44	1	1	1	0.98	1	1	1	1	1	1	1	1	1	1
45	1	1	1	1	0.99	1	1	1	1	1	1	1	1	1
46	1	1	1	1	1	1.00	1	1	1	1	1	1	1	1
48	1	1	1	1	1	1	0.99	1	1	1	1	1	1	1
49	1	1	1	1	1	1	1	1	1	1	1	1	1	1
50	1	1	1	1	1	1	0.97	1	0.97	1	1	1	1	1
51	1	1	1	1	1	1	1	1	1	0.99	1	1	1	1
52	1	1	1	1	1	1	1	1	1	1	0.93	1	1	1
53	1	1	1	1	1	1	1	1	1	1	1	0.96	1	1
54	1	1	1	1	1	1	1	1	1	1	1	1	0.97	1
55	1	1	1	1	1	1	1	1	1	1	1	1	1	0.98
56	1	1	1	1	1	1	1	1	1	1	1	1	1	1
57	1	1	1	1	1	1	1	1	1	1	1	1	1	1
59	1	1	1	1	1	1	1	1	1	1	1	1	1	1
60	1	1	1	1	1	1	1	1	1	1	1	1	1	1
61	1	1	1	1	1	1	1	1	1	1	1	1	1	1
63	1	1	1	1	1	1	1	1	1	1	1	1	1	1
64	1	1	1	1	1	1	1	1	1	1	1	1	1	1
65	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Table 2. The confusion matrix of validation test (Continued 3)

Regime	56	57	59	60	61	63	64	65	
2	1	1	1	1	1	1	1	1	1
3	1	1	1	1	1	1	1	1	1
4	1	1	1	1	1	1	1	1	1
5	1	1	1	1	1	1	1	1	1.00
7	1	1	1	1	1	1	1	1	1
8	1	1	1	1	1	1	1	1	1
9	1	1	1	1	1	1	1	1	1
10	1	1	1	1	1	1	1	1	1.00
11	1	1	1	1	1	1	1	1	1.00
12	1	1	1	1	1	1	1	1	1.00
13	1	1	1	1	1	1	1	1	1
14	1	1	1	1	1	1	1	1	1.00
15	1	1	1	1	1	1	1	1	1
16	1	1	1	1	1	1	1	1	1.00
17	1	1	1	1	1	1	1	1	1
19	1	1	1	1	1	1	1	1	1
20	1	1	1	1	1	1	1	1	1
21	1	1	1	1	1	1	1	1	1
22	1	1	1	1	1	1	1	1	1
23	1	1	1	1	1	1	1	1	1
24	1	1	1	1	1	1	1	1	1
25	1	1	1	1	1	1	1	1	1
26	1	1	1	1	1	1	1	1	1
27	1	1	1	1	1	1	1	1	1.00
28	1	1	1	1	1	1	1	1	1.00
36	1	1	1	1	1	1	1	1	1
37	1	1	1	1	1	1	1	1	1
40	1	1	1	1	1	1	1	1	1
41	1	1	1	1	1	1	1	1	1
42	1	1	1	1	1	1	1	1	1
43	1	1	1	1	1	1	1	1	1
44	1	1	1	1	1	1	1	1	1
45	1	1	1	1	1	1	1	1	1
46	1	1	1	1	1	1	1	1	1
48	1	1	1	1	1	1	1	1	1
49	1	1	1	1	1	1	1	1	1
50	1	1	1	1	1	1	1	1	1.00

Table 2. The confusion matrix of validation test (Continued 4)

Regime	56	57	59	60	61	63	64	65	
51	1	1	1	1	1	1	1	1	1
52	1	1	1	1	1	1	1	1	1.00
53	1	1	1	1	1	1	1	1	1.00
54	1	1	1	1	1	1	1	1	1.00
55	1	1	1	1	1	1	1	1	1
56	0.97	1	1	1	1	1	1	1	1.00
57	1	0.98	1	1	1	1	1	1	1
59	1	1	1.00	1	1	1	1	1	1
60	1	1	1	0.99	1	1	1	1	1
61	1	1	1	1	0.99	1	1	1	1
63	1	1	1	1	1	0.99	1	1	1
64	1	1	1	1	1	1	0.98	1	1
65	1	1	1	1	1	1	1	0.98	1
Overall Accuracy									**0.99**

Table 2. The confusion matrix of validation test (Continued 5)

In addition to the validation test, the performance of the HMM based regime recognition algorithm was compared with a number of data mining methods. These data mining methods included: neural network, discriminant analysis, K-nearest neighbor, regression tree, and naïve bayes. The results of the performance comparison test are provided in Table 3. In this test, to be consistent, data with the same regimes were used for all the data mining methods. From Table 3, we can see that the HMM based regime recognition algorithm outperforms all other data mining methods.

Note that in Table 3, the names of methods are defined as: HMM = hidden Markov model; NN = neural network (back propagation); DA = discriminant analysis; KNN = k-nearest neighbor; RT = regression tree; NB = naïve bayes.

4. Conclusion

In this paper, a data mining approach is adopted for regime recognition. In particular, a regime recognition algorithm developed based on HMM was presented. The HMM based regime recognition involves two major stages: model learning process and model testing process. The learning process could be implemented off-board. In this process, Gaussian mixture model (GMM) was used instead of unimodal density of Gaussian distribution in HMM. Once the learning process is completed, new incoming unknown signal could be tested and recognized on-board. The developed algorithm was validated using the flight card data of an Army UH-60L helicopter. The performance of this regime recognition algorithm was also compared with other data mining approaches using the same dataset. Using the flight card information and regime definitions, a dataset of about 56,000 data points labeled with 50 regimes recorded in the flight card were mapped to the health and usage monitoring parameters. The validation and performance comparison results have showed that the hidden Markov model based regime recognition algorithm was able to accurately recognize the regimes recorded in the flight card data and outperformed other data mining methods.

Regime No.	HMM	NN	DA	KNN	RT	NB
2	0.01%	32.00%	16.00%	0.00%	12.00%	46.00%
3	0.10%	17.65%	23.53%	41.18%	100.00%	94.12%
4	0.06%	0.00%	84.62%	69.23%	100.00%	84.62%
5	10.30%	3.03%	0.00%	0.00%	39.39%	75.76%
7	2.51%	10.20%	5.10%	10.20%	3.06%	0.00%
8	0.26%	10.00%	6.67%	43.33%	0.00%	53.33%
9	0.04%	18.75%	62.50%	56.25%	100.00%	100.00%
10	0.66%	50.00%	50.00%	35.71%	100.00%	100.00%
11	1.02%	0.00%	26.67%	20.00%	13.33%	66.67%
12	0.92%	0.00%	0.00%	0.00%	100.00%	88.89%
13	0.05%	0.00%	0.00%	3.45%	100.00%	68.97%
14	0.21%	13.33%	6.67%	46.67%	100.00%	80.00%
15	0.09%	33.33%	33.33%	66.67%	100.00%	50.00%
16	0.88%	40.00%	100.00%	100.00%	100.00%	80.00%
17	0.12%	100.00%	100.00%	100.00%	100.00%	100.00%
19	0.11%	10.00%	0.00%	50.00%	100.00%	80.00%
20	0.09%	0.00%	0.00%	0.00%	0.00%	70.83%
21	0.09%	9.09%	36.36%	36.36%	100.00%	100.00%
22	0.01%	21.43%	14.29%	50.00%	100.00%	100.00%
23	0.07%	0.00%	9.09%	9.09%	100.00%	90.91%
Overall	0.88%	12.79%	15.58%	21.16%	47.44%	57.44%

Table 3. Results of performance comparison of various data mining methods (regime recognition error rate)

5. References

Berry, J., Vaughan, R., Keller, J., Jacobs, J., Grabill, P., and Johnson, T. (2006). Automatic Regime Recognition using Neural Networks. *Proceedings of American Helicopter Society 60th Annual Forum,* Phoenix, AZ, May 9 – 11, 2006.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society*, Vol. 39, No. 1, pp. 1 - 38.

Le, D. (2006). Federal Aviation Administration (FAA) Health and Usage Monitoring System R&D Initiatives. http://aar400.tc.faa.gov/Programs/agingaircraft/rotorcraft/index.htm.

Levinson, S. E., Rabiner, L. R., and Sondhi, M. M. (1983). An Introduction to the Application of the Theory of Probabilistic Functions of a Markov Process to Automatic Speech Recognition. *Bell System Technical Journal*, Vol. 62, No. 4, pp. 1035 - 1074.

Rabiner, L. R. (1989). A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE*, Vol. 77, No. 2, pp. 257 - 286.

Teal, R. S., Evernham, J. T., Larchuk, T. J., Miller, D. G., Marquith, D. E., White, F., and Deibler, D. T. (1997). Regime Recognition for MH-47E Structure Usage Monitoring. *Proceedings of American Helicopter Society 53rd Annual Forum,* Virginia Beach, VA, April 29 – May 1, 1997.

Aerospace Technologies Advancements

Edited by Thawar T. Arif

ISBN 978-953-7619-96-1
Hard cover, 492 pages
Publisher InTech
Published online 01, January, 2010
Published in print edition January, 2010

Space technology has become increasingly important after the great development and rapid progress in information and communication technology as well as the technology of space exploration. This book deals with the latest and most prominent research in space technology. The first part of the book (first six chapters) deals with the algorithms and software used in information processing, communications and control of spacecrafts. The second part (chapters 7 to 10) deals with the latest research on the space structures. The third part (chapters 11 to 14) deals with some of the latest applications in space. The fourth part (chapters 15 and 16) deals with small satellite technologies. The fifth part (chapters 17 to 20) deals with some of the latest applications in the field of aircrafts. The sixth part (chapters 21 to 25) outlines some recent research efforts in different subjects.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821